

# **Лабораторная работа №1. Определение параметров сетевого соединения компьютера**

## **Введение**

Сейчас любая локальная сеть, как правило, имеет подключение к глобальной сети Интернет. В Интернет передача данных осуществляется с помощью протоколов TCP/IP (Transmission Control Protocol/Internet Protocol) и в локальной сети эти протоколы оказываются также необходимы. Но использование протоколов стека TCP/IP позволяет решать и задачи обмена информацией между локальными компьютерами. Применение в локальных сетях протоколов TCP/IP сближает их с глобальными компьютерными сетями в смысле использования подобных способов адресации и методов администрирования.

## **IP-адресация**

Передача сообщений в Интернет основана на том, что каждый компьютер сети имеет индивидуальный адрес – IP-адрес. Этот адрес выражается одним 32-разрядным числом, имеющим две смысловые части. Одна часть IP-адреса определяет номер сети, вторая – номер узла (компьютера) в сети. Так как оперировать длинными двоичными числами достаточно сложно, число, определяющее IP-адрес, разбивают на 4 октета – восьмиразрядных двоичных числа, а каждое из этих чисел представляют в десятичном виде. Октеты отделяют друг от друга точками. Таким образом, 32-разрядный IP-адрес представляется в виде: 255.255.255.255 (десятичное число может меняться от 0 до 255 – максимального значения восьмиразрядного двоичного числа). Например: 128.10.2.30 – десятичная форма представления IP-адреса, 10000000 00001010 00000010 00011110 – двоичная форма представления этого же адреса.

В сети Интернет различные глобальные сети, в зависимости от размера,

делятся по классам:

- Сети класса А: большие сети общего пользования, первый октет определяет номер сети, три последующие октета – номер узла;
- Сети класса В: сети среднего размера. Два первых октета определяют номер сети, два оставшихся – номер узла;
- Сети класса С: сети малого размера. В этих сетях три первых октета определяют номер сети и последний октет – номер узла.

В таблице 1 представлена общая характеристика схемы Интернет-адресации.

Таблица 1

Класс	Диапазон значений первого октета	Общее количество сетей	Максимальное количество узлов в каждой сети
А	1 – 126	126	16 777 214
В	128 – 191	16 382	65 534
С	192 – 223	2 097 150	254

Некоторые IP-адреса имеют специальное назначение, например, адрес:

- 0.0.0.0 представляет адрес шлюза по умолчанию, т.е. адрес компьютера, которому следует направлять информационные пакеты, если они не нашли адресата в локальной сети;
- 127.любое число (часто 127.0.0.1) – адрес «петли». Данные, переданные по этому адресу, поступают на вход компьютера, как полученные по сети. Такой адрес необходим при отладке сетевых программ;
- 255.255.255.255 – широковещательный адрес. Сообщения, переданные по этому адресу, получают все узлы локальной сети, содержащей компьютер-источник сообщения (в другие локальные сети оно не передается);
- Номер сети . все нули – адрес сети;
- Все нули . номер узла – узел в данной сети. Может использоваться для передачи сообщений конкретному узлу внутри локальной сети;
- Номер сети . все единицы (двоичные) – все узлы указанной сети.

В локальных сетях используются специальные, так называемые «серые» IP-адреса. Они определены документом RFC 1918 (RFC – Requests For Comments, предлагаемый проект стандарта, большинство документов, регламентирующих Интернет, описано в RFC) и приведены в табл. 2:

Таблица 2

<b>Диапазоны IP-адресов, используемых в локальных сетях</b>
10.0.0.0 – 10.255.255.255
172.16.0.0 – 172.31.255.255
192.168.0.0 – 192.168.255.255

В небольших по размеру локальных сетях обычно применяется последний диапазон адресов. Сетевые маршрутизаторы не передают информацию для узлов с этими адресами, поэтому она оказывается «запертой» внутри локальной сети. Такая схема позволяет в разных локальных сетях использовать одни и те же IP-адреса и не приводит к конфликтам.

Для повышения гибкости использования IP-адресов деление адреса на части с использованием классов дополняется технологией CIDR (Classless Inter-Domain Routing) – бесклассовой междоменной маршрутизации. В этом случае адрес сети формируется с помощью двух чисел: адреса и **маски**. Маска это тоже 32-разрядное двоичное число, с помощью которого из IP-адреса выделяется адрес сети. Схема формирования адреса сети с использованием маски проста, ее можно пояснить на примере, допустим, адрес представлен двоичным числом 110101, маска числом 111100. Маска накладывается на адрес, как трафарет, в котором единицы соответствуют прорезам, в которых мы «увидим» адрес сети, в нашем примере адрес сети соответствует числу 110100. Маска всегда содержит такое двоичное число, старшие разряды которого подряд единицы, а младшие – нули, единицы представляют «прозрачную» часть трафарета, а нули – «непрозрачную». Маска так же, как и адрес, записывается в виде четырех десятичных чисел, разделенных точками и представляющих двоичные октеты. Для компактной записи пары чисел: IP-адрес-маска, используется также другая

форма, например: 10.0.0.8/30. Число до слеша представляет собой IP-адрес, а число после слеша – количество разрядов в IP-адресе, отводимых для адресации сети. Число 30 после слеша соответствует маске 255.255.255.252. После определения адреса сети, оставшаяся часть IP-адреса используется для адресации узлов в сети.

### **Символьное представление имени компьютера в сети**

Каждый компьютер в сети имеет уникальный адрес. При использовании IP-адресации это IP-адрес. Однако человеку достаточно трудно оперировать длинными наборами цифр, не несущих смысловой нагрузки, поэтому всегда применяются системы преобразования имен, ставящие в соответствие цифровому адресу компьютера его символьное имя. В глобальных сетях и сети Интернет это служба DNS (Domain Name System) – распределенная база данных, поддерживающая иерархическую систему имен для идентификации узлов в сети Интернет. Определенные части базы данных доменных имен хранятся на специальных серверах – DNS-серверах, обрабатывающих запросы любого компьютера и определяющие имя, соответствующее IP-адресу или наоборот. В каждой локальной сети, подключенной к Интернет, работает по крайней мере один DNS-сервер. База данных DNS имеет структуру дерева, называемого доменным пространством имен, в котором каждый домен (узел дерева) имеет имя и может содержать поддомены. Имя домена идентифицирует его положение в этой базе данных по отношению к родительскому домену, а точки в имени отделяют части, соответствующие узлам домена, например, [www.tusur.ru](http://www.tusur.ru).

Для именования компьютеров в локальных сетях используются плоские (не имеющие иерархии) символьные имена, так называемые NetBIOS-имена. Протокол NetBIOS (Network Basic Input/Output System), как расширение стандартных функций базовой системы ввода-вывода, был разработан в 1984г. компанией IBM и широко применяется в ее продуктах, а также продуктах компании Microsoft. В протоколе NetBIOS реализован механизм широковеща-

тельного разрешения имен, когда все компьютеры в локальной сети получают запрос на разрешение имени, соответствующего некоторому IP-адресу. Кроме того, компания Microsoft для своей сетевой операционной системы Windows NT разработала централизованную службу разрешения имен WINS(Windows Internet Name Service). WINS-сервер, работающий в локальной сети, централизованно обрабатывает все запросы, касающиеся разрешения имен в сетях Windows. При большом числе компьютеров в локальной сети WINS-сервер необходим. Однако в малых сетях, содержащих менее 10 компьютеров, часто используется широковещательный механизм разрешения имен протокола NetBIOS, упрощающий административное обслуживание таких сетей.

### **Автоматизация процесса назначения IP-адресов узлам сети**

IP-адреса могут назначаться администратором сети вручную. Это представляет для администратора достаточно сложную и длительную процедуру, если количество компьютеров в локальной сети достаточно велико. Если происходят изменения в сети, например, появляются новые компьютеры, процедуру необходимо выполнить и для них, а в некоторых случаях и выполнить коррекцию предыдущих настроек на уже работающих компьютерах. Протокол DHCP (Dynamic Host Configuration Protocol) был разработан для того, чтобы освободить администратора от этих проблем. Основным назначением DHCP является динамическое назначение IP-адресов. В локальной сети, содержащей DHCP-сервер, каждый компьютер при включении посылает запрос этому серверу на получение IP-адреса. Способы выдачи адресов могут быть различными.

При автоматическом статическом способе выделения адреса DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула (набора) наличных IP-адресов. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Между идентификатором клиента и его IP-адресом в этом случае, как и при ручном назначении, существует постоянное соответствие.

При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время (время подключения к сети), что дает возможность впоследствии повторно использовать этот же IP-адрес другими компьютерами (пользователями).

### **Адресация компьютеров на канальном уровне**

Каждый компьютер, подключенный к сети, имеет сетевой адаптер (сетевую карту) с присвоенным ему адресом. Этот адрес носит название MAC-адреса, он задается при изготовлении сетевого адаптера и впоследствии не изменяется. Длина и другие особенности MAC-адреса зависят от используемой в локальной сети технологии. В сетях Ethernet MAC-адрес имеет длину 6 байт, записанных в шестнадцатеричном формате и разделенных дефисами (например 00-AA-00-4F-2A-9C). Для определения локального адреса по IP-адресу используется протокол разрешения адреса ARP (Address Resolution Protocol). Существует также протокол, решающий обратную задачу – нахождение IP-адреса по известному локальному адресу. Он называется RARP – реверсивный ARP, и используется при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих адрес своего сетевого адаптера.

Необходимость в обращении к протоколу ARP возникает каждый раз, когда модуль IP передает пакет на уровень сетевых интерфейсов, например драйверу Ethernet. IP-адрес узла назначения известен модулю IP. Требуется на его основе найти MAC-адрес узла назначения. Работа протокола ARP начинается с просмотра так называемой ARP-таблицы (рис.). Каждая строка таблицы устанавливает соответствие между IP-адресом и MAC-адресом. Поле «Тип записи» может содержать одно из двух значений – «динамический» или «статический». Статические записи создаются вручную с помощью утилиты **arp** и не имеют срока устаревания, точнее, они существуют до тех пор, пока компьютер или маршрутизатор не будут выключены. Динамические же записи создаются модулем протокола ARP, использующим широковещательные возможности ло-

кальных сетевых технологий. Динамические записи должны периодически обновляться. Если запись не обновлялась в течение определенного времени (порядка нескольких минут), то она исключается из таблицы. Таким образом, в ARP-таблице содержатся записи не обо всех узлах сети, а только о тех, которые активно участвуют в сетевых операциях. Поскольку такой способ хранения информации называют кэшированием, ARP-таблицы иногда называют ARP-кэш. После того как модуль IP обратился к модулю ARP с запросом на разрешение адреса, происходит поиск в ARP-таблице указанного в запросе IP-адреса. Если таковой адрес в ARP-таблице отсутствует, то исходящий IP-пакет, для которого нужно было определить локальный адрес, ставится в очередь. Далее протокол ARP формирует свой запрос (ARP-запрос), вкладывает его в кадр протокола канального уровня и рассылает запрос широковещательно. Все узлы локальной сети получают ARP-запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой локальный адрес, а затем отправляет его уже по адресу компьютера, сформировавшего запрос, так как в адрес отправителя указан в самом запросе.

## **Сетевые утилиты**

В операционной системе Linux существует большое число утилит (специальных программ), предназначенных для управления и анализа сетевых соединений, рассмотрим три из них: IFCONFIG, ARP, NETSTAT.

### **Утилита IFCONFIG**

Позволяет просмотреть текущую конфигурацию адресов TCP/IP для всех установленных на данном компьютере сетевых адаптеров и коммутируемых соединений, с ее помощью можно определить IP-адрес данного компьютера. Запущенная без параметров, команда ifconfig выдает в качестве результата текущую конфигурацию адресов TCP/IP для всех установленных на данном компьютере

сетевых адаптеров и коммутируемых соединений:

```
eth0      Link encap:Ethernet  HWaddr 00:1d:60:38:94:4f
          inet addr:192.168.0.130  Bcast:192.168.0.255
Mask:255.255.255.0
          inet6 addr: fe80::21d:60ff:fe38:944f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:246109 errors:0 dropped:0 overruns:0 frame:0
          TX packets:211095 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:224796147 (224.7 MB)  TX bytes:40813580 (40.8 MB)
          Interrupt:27 Base address:0xa000

lo        Link encap:Локальная петля (Loopback)
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:12 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:720 (720.0 B)  TX bytes:720 (720.0 B)

wlan0     Link encap:Ethernet  HWaddr 00:1b:77:72:2d:6e
          inet6 addr: fe80::21b:77ff:fe72:2d6e/64 Scope:Link
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:1075 errors:0 dropped:0 overruns:0 frame:0
          TX packets:121 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:86321 (86.3 KB)  TX bytes:26876 (26.8 KB)
```

Команду `ifconfig` следует первой использовать для диагностирования возможных проблем с соединением TCP/IP. С ее помощью можно определить, были ли вообще назначен IP-адрес сетевому адаптеру, а также узнать адрес шлюза.



## Утилита NETSTAT

Команда позволяет получить подробную информацию о соединениях, активных в настоящее время. Дополнительные ключи позволяют также получить информацию о сетевых портах, об IP-адресах компьютеров, участвующих в подключении, а также о других сетевых параметрах.

Параметры:

**-a**

Вывод всех активных подключений TCP и прослушиваемых компьютером портов TCP и UDP (рис.).

**-e**

Вывод статистики Ethernet, например количества отправленных и принятых байтов и пакетов. Этот параметр может комбинироваться с ключом **-s**.

**-n**

Вывод активных подключений TCP с отображением адресов и номеров портов в числовом формате без попыток определения имен.

```
netstat -a
```

Активные соединения с интернетом (servers and established)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	localhost:ipp	*:*	LISTEN
tcp	0	0	*:17500	*:*	LISTEN
tcp	0	0	u5f.local:33799	sjc-not19.sjc.dropb:www	ESTABLISHED
tcp	38	0	u5f.local:60223	75.126.110.108-st:https	CLOSE_WAIT
tcp	38	0	u5f.local:34882	v-client-2b.sjc.d:https	CLOSE_WAIT
tcp	38	0	u5f.local:40990	v-client-1b.sjc.d:https	CLOSE_WAIT
tcp	38	0	u5f.local:34500	ec2-50-16-225-86.:https	CLOSE_WAIT
tcp	38	0	u5f.local:38920	v-d-1a.sjc.dropbo:https	CLOSE_WAIT
tcp	38	0	u5f.local:57300	ec2-50-16-213-132:https	CLOSE_WAIT
tcp6	0	0	localhost:ipp	[::]:*	LISTEN
udp	0	0	*:bootpc	*:*	
udp	0	0	*:17500	*:*	
udp	0	0	*:mdns	*:*	
udp	0	0	*:41844	*:*	

**-o**

вывод активных подключений TCP и включение кода процесса (PID) для каждого подключения. Код процесса позволяет найти приложение на вкладке **Процессы** диспетчера задач Windows. Этот параметр может комбинироваться с ключами **-a**, **-n** и **-p**.

**-p** *протокол*

Вывод подключений для протокола, указанного параметром *протокол*. В этом случае параметр *протокол* может принимать значения **tcp**, **udp**, **tcpv6** или **udpv6**. Если данный параметр используется с ключом **-s** для вывода статистики по протоколу, параметр *протокол* может иметь значение **tcp**, **udp**, **icmp**, **ip**, **tcpv6**, **udpv6**, **icmpv6** или **ipv6**.

**-s**

Вывод статистики по протоколу. По умолчанию выводится статистика для протоколов TCP, UDP, ICMP и IP. Если установлен протокол IPv6 для Windows XP, отображается статистика для протоколов TCP через IPv6, UDP через IPv6, ICMPv6 и IPv6. Параметр **-p** может использоваться для указания набора протоколов.

**-r**

Вывод содержимого таблицы маршрутизации IP:

```
netstat -r
```

Таблица маршрутизации ядра протокола IP

Destination	Gateway	Genmask	Flags	MSS	Window	irrt	Iface
192.168.0.0	*	255.255.255.0	U	0	0	0	eth0
link-local	*	255.255.0.0	U	0	0	0	eth0
default	192.168.0.1	0.0.0.0	UG	0	0	0	eth0

*интервал*

Обновление выбранных данных с интервалом, определенным параметром *интервал* (в секундах). Нажатие клавиш CTRL+C останавливает обновление. Если этот параметр пропущен, **netstat** выводит выбранные данные только один раз.

/? Отображение справки в командной строке.

## Утилита ARP

Служит для вывода и изменения записей кэша протокола ARP, который содержит одну или несколько таблиц, использующихся для хранения IP-адресов и соответствующих им физических адресов Ethernet или Token Ring. Для каждого сетевого адаптера Ethernet или Token Ring, установленного в компьютере, используется отдельная таблица. Запущенная без параметров, команда `arp` выводит справку.

### Параметры

**-a**

Вывод таблиц текущего протокола ARP для всех интерфейсов

Чтобы вывести записи ARP для определенного IP-адреса, следует указать его после ключа через пробел:

**Arp -a IP-адрес**

Чтобы вывести таблицы кэша ARP для определенного интерфейса, следует указать параметр **-N**

**Arp -a -N иф\_адрес**

*иф\_адрес*, где *иф\_адрес* – это IP-адрес, назначенный интерфейсу. Параметр **-N** вводится с учетом регистра.

**-g** Выполняет те же функции, что и **-a**.

**-d IP-адрес [иф\_адрес]**

Выполняет удаление записи с определенным IP-адресом. Чтобы удалить запись таблицы для определенного интерфейса, следует указать этот интерфейс после IP-адреса. Чтобы удалить все записи, нужно ввести звездочку (\*) вместо параметра *IP-адрес*.

**-s IP-адрес Ethernet\_адрес [иф\_адрес]**

Добавление статической записи, которая сопоставляет IP-адрес с физическим адресом в кэш ARP. Чтобы добавить статическую запись кэша ARP в таблицу

для определенного интерфейса, следует указать параметр *иф\_адрес*, где *иф\_адрес* – это IP-адрес, назначенный интерфейсу.

## Задание на лабораторную работу

Работа выполняется индивидуально. С помощью утилит IFCONFIG, ARP, NETSTAT необходимо получить информацию для заполнения таблиц 3-5.

Табл. 3

Символьное имя компью- тера	Адрес локаль- ной сети	IP-адрес компьютера	MAC-адрес компьютера	Используй- мая в ло- кальной сети технология

Табл. 4

Таблица маршрутизации. Активные маршруты:				
Сетевой ад- рес	Маска подсети	Адрес шлюза	Интерфейс	Метрика

Табл. 5

Таблица ARP-кэша:		
IP-адрес	MAC-адрес	Тип

Кроме этого, необходимо определить используются ли в локальной сети серверы DNS, WINS, DHCP и если используются, указать их IP-адреса.

## Содержание отчета

Отчет по проделанной работе готовится в текстовом редакторе OpenOffice.org Write и предоставляется для проверки в электронном виде в формате электронных документов PDF.

Отчет должен состоять из следующих частей:

- введение

- постановка задачи
- основная часть
- заключение
- приложение

## **Лабораторная работа №2. Использование сетевых утилит операционной системы**

### **Введение**

#### **Команда PING**

Команда PING является едва ли не самой используемой в локальных сетях командой. Она позволяет тестировать сетевое соединение, получая информацию о различных его аспектах. Неудачная попытка соединения с каким-либо компьютером, или ошибка получения доступа к общим файлам и папкам, находящимся на других компьютерах локальной сети, может быть вызвана тем, что другие компьютеры просто не получают отправленных им по сети запросов. После введения в командной строке имени команды, в качестве параметра для нее, указывается адрес по которому будут направляться специальные эхо-пакеты, это может быть IP-адрес (рис. 1), или символьное имя компьютера. Получив эхо-запрос, удаленный компьютер сразу же отправляет его обратно по тому адресу, откуда он пришел, команда ping позволяет узнать, пришли ли обратно посланные запросы, проверяя таким образом не только целостность физической среды передачи данных, но и корректную обработку информации на всех остальных семи уровнях модели OSI.

```
ping -c 3 www.tusur.ru
```

```
PING www.tusur.ru (88.204.75.138) 56(84) bytes of data.
```

```
64 bytes from portal.tusur.ru (88.204.75.138): icmp_seq=1 ttl=60
```

```
time=0.860 ms
```

```
64 bytes from portal.tusur.ru (88.204.75.138): icmp_seq=2 ttl=60
time=0.909 ms
64 bytes from portal.tusur.ru (88.204.75.138): icmp_seq=3 ttl=60
time=0.873 ms
```

```
--- www.tusur.ru ping statistics ---
```

```
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.860/0.880/0.909/0.040 ms
```

При успешном возвращении запросов можно быть уверенным в том, что среда передачи данных, программное обеспечение ТСП/IP, а также все устройства (маршрутизаторы, повторители и др.), встретившиеся на пути между двумя компьютерами, работают нормально. Необходимо отметить, что даже при отсутствии каких-либо неисправностей на пути между двумя компьютерами, один или сразу несколько пакетов могут быть утеряны, как правило, это бывает в случае перегруженности сети, а также с тем, что диагностирующие пакеты имеют очень низкий приоритет и могут быть отброшены в процессе передачи. Если хотя бы один из посланных пакетов вернется, это уже будет означать исправность работы сети. По-умолчанию размер эхо-пакета составляет 32 байта, по указанному адресу направляются эхо-пакеты и после выполнения команды выводится статистика прохождения эхо-пакетов по сети.

## **Команда TRACEROUTE**

Эта команда подобна команде PING, обе посылают в точку назначения эхо-пакеты и затем ожидают их возвращения. Отличие пакетов команды TRACEROUTE от пакетов PING заключается в том, что они имеют различный срок жизни (Time to Live, TTL). Каждый маршрутизатор при прохождении через него пакета уменьшает значение поля TTL в нем на единицу. Первые пакеты, отправляемые командой TRACEROUTE имеют TTL=1, поэтому первый маршрутизатор, получив такой пакет и уменьшив на единицу поле TTL, обнаруживает, что пакет не может быть доставлен по адресу (пакет с TTL=0 не передается маршрутизатором) и возвращает сообщение об ошибке, содержащее IP-адрес

маршрутизатора. Получив это сообщение, команда выводит на экран информацию об IP-адресе маршрутизатора и отправляет по прежнему адресу эхо-пакет с TTL=2. Количество маршрутизаторов, через которые может пройти пакет, будет каждый раз увеличиваться на единицу до тех пор, пока пакет не достигнет точки назначения. Таким образом, с помощью команды traceroute можно получить подробный маршрут прохождения пакетов данных между компьютером, на котором была запущена traceroute, и любым удаленным компьютером сети. Это делает traceroute весьма ценным средством обнаружения неисправностей в сетевом соединении: в случае возникновения проблемы с подключением к Web-узлу или к какой-нибудь другой службе Internet можно определить участок, на котором она возникла.

```
traceroute ed.tusur.ru
```

```
traceroute to ed.tusur.ru (88.204.77.199), 30 hops max, 60 byte packets
```

```
 1  192.168.0.1 (192.168.0.1)  0.404 ms  0.586 ms  0.706 ms
 2  fet-internal.fet.tusur.ru (212.192.122.225)  5.500 ms  5.664 ms
5.878 ms
 3  cl-ft.SUR.net.ru (88.204.72.249)  3.658 ms  3.631 ms  3.718 ms
 4  dl-rk-4.SUR.net.ru (88.204.72.218)  3.688 ms  3.912 ms  4.001 ms
 5  ed.tusur.ru (88.204.77.199)  3.731 ms  3.698 ms  3.856 ms
```

## Утилита NSLOOKUP

Утилита nslookup (англ. name server lookup поиск на сервере имён) — утилита, предоставляющая пользователю интерфейс командной строки для обращения к системе DNS (проще говоря, DNS-клиент). Позволяет задавать различные типы запросов и запрашивать произвольно указываемые сервера.

```
nslookup www.tusur.ru
```

```
Server:          212.192.122.17
```

```
Address:  212.192.122.17#53
```

```
Name: www.tusur.ru
```

```
Address: 88.204.75.138
```



## ***Цель работы***

Изучить работу сетевых утилит операционной системы Linux.

## ***Задание на лабораторную работу***

Используя сетевые утилиты PING, TRACEROUTE и NSLOOKUP исследовать свойства сетевых соединений компьютера.

## ***Порядок выполнения работы***

С помощью утилиты PING протестировать соединения с серверами находящимися на разном "расстоянии" от нас: в локальной сети ТУСУР, в городской томской сети, в российском сегменте Интернет, в "мировом" интернете.

С помощью утилиты TRACEROUTE протестировать соединения с серверами находящимися на разном "расстоянии" от нас: в локальной сети ТУСУР, в городской томской сети, в российском сегменте Интернет, в "мировом" интернете.

С помощью утилиты NSLOOKUP определить IP-адреса нескольких интернет ресурсов.

По итогам выполнения работы подготовить отчет. В отчете необходимо представить результаты работы утилит.